

Sun Java™ System Application Server Enterprise Edition Release Notes for Microsoft Windows

Version 8.1 2005Q2

Part Number 819-4264-10

The Sun Java System Application Server Enterprise Edition 8.1 2005Q2 product greatly simplifies the task of creating and administering web services applications. It provides superior performance, clustering, and high availability features for scalable services that continue to operate despite software and hardware faults. The Application Server provides a development path for web services that simplifies the development process while providing uniquely flexible growth opportunities.

These release notes contain important information available at the time of the Sun Java System Application Server 8.1 2005Q2 product release for Windows. Component requirements, platform summary, known problems, and other late-breaking issues are addressed here. Read this document before you begin using the Application Server product.

The most up-to-date version of these release notes can be found at the Sun Java System documentation web site: <http://docs.sun.com/app/docs/prod/entsys.05q4#hic>. Check the web site prior to installing and setting up your software and then periodically thereafter to view the most up-to-date release notes and product documentation.

This document contains the following sections:

- [Release Notes Revision History](#)
- [About Application Server 8.1 2005Q2](#)
- [Bugs Fixed in This Release](#)
- [Important Information](#)
- [Known Issues and Limitations](#)
- [Redistributable Files](#)
- [How to Report Problems and Provide Feedback](#)
- [Additional Sun Resources](#)

Third-party URLs are referenced in this document and provide additional, related information.

NOTE Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Release Notes Revision History

This section lists the changes that have been made in these release notes after the initial release of the Application Server 2005Q2 component.

Table 1 Revision History

Revision Date	Description
February 2006	Revenue release.
November 2005	Beta release.

About Application Server 8.1 2005Q2

The Sun Java System Application Server Enterprise Edition 8.1 is a J2EE 1.4 platform-compatible server for the development and deployment of J2EE applications and Java technology-based web services in large-scale production environments.

This section includes:

- [What's New in Application Server 8.1 2005Q2](#)
- [Hardware and Software Requirements](#)
- [Related Documentation](#)

What's New in Application Server 8.1 2005Q2

The Application Server Enterprise Edition 8.1 2005Q2 Update 2 includes the following enhancements:

- **Improved Administration** — The Application Server supports the remote secure management of complex multi-machine enterprise deployments using either a browser based console or a scriptable command line interface. It also provides a rich JMX based API allowing remote, secure, programmatic access to administrative and monitoring functions.
- **Message Broker** — The Application Server is bundled with an integrated enterprise class message broker that features providing highly available, reliable, high performance, and scalable messaging.
- **Expanded Platform Support** — Additional operating systems, databases, locales, and hardware are supported.
- **Sun Java Enterprise System** — As a key component of the Sun Java Enterprise System, the Application Server is tightly integrated with portal and network identity services.
- **Migration and Upgrade Tools** — These tools enable you to verify J2EE applications for standards conformance and portability, help with migrations from other J2EE Application Servers (JBoss, WebLogic, WebSphere), and aid in upgrading from previous versions of Sun ONE Application Server/ iPlanet Application Server.
- **Java 2 Standard Edition 5.0 Support** — The Application Server supports the Java 2 Standard Edition 5.0, which includes enhanced management and monitoring features and many performance and scalability improvements.
- **Java Web Services Developer Pack 1.6 (JWSDP) Plugin Support** — All JWSDP plugins are now supported. The JWSDP 1.6 can be downloaded for free from the Java Download Center (<http://java.sun.com/webservices/downloads/webservicespack.html>).
- **JDBC Drivers** — The Application Server is bundled with Sun JDBC drivers.
- **Web Services Security** — These container message security mechanisms implement message-level authentication (for example, XML digital signature and encryption) of SOAP web services invocations using the X509 and username/password profiles of the OASIS WS-Security standard.
- **WS-I Basic Profile 1.1** — As mandated by the J2EE 1.4 specification, this release implements Web Services Interoperability (WS-I) Basic Profile 1.1 to enable interoperability for web services applications.

- **Backend Connectivity with iWay Adapters** — Sun Microsystems now resells and supports twenty-two iWay adapters to key backend systems (SAP, Siebel, Oracle, CICS, and IBM MQ Series) to help you leverage existing IT applications from within the Application Server environment. These adapters support the J2EE Connector Architecture 1.5 specification and Web services (SOAP) standards, and include developer tools to reduce time to connect to backend applications.
- **Latest HADB Management System** — Sun Java Enterprise System contains the new High Availability Data Base (HADB version 4.4-2.7). See the High Availability Guide for the details on HADB requirement and limitations.
- A new management command `hadbm setadminpassword` has been implemented to allow changing the password used for database administration. The command takes options indicating which management agent to use, and the old and new password. For more information, see the `hadbm setadminpassword` manpage.
- The existing management command `hadbm listpackages` has been modified. Previously, the command took no operands, and listed all packages in the relevant management domain. The modifications introduces an optional package name operand, and lists only packages with that name. If the operand is not provided, all packages are listed. For more information, see the `hadbm listpackages` manpage.
- The existing management command `hadbm createdomain` has been modified. The `hostlist` operand is extended to also specify the port number of the management agent. In this way, the domain is completely specified using only the `hostlist` operand. The old behavior is still supported for backward compatibility. For more information, see the `hadbm createdomain` manpage.
- Some of the error messages from the management system have been modified. The modifications are intended to improve understandability, consistency and accuracy of the error messages. The actual modifications are not listed in these release notes.

Application Server Product Releases

The Application Server product is delivered in various ways. The following table identifies the product delivered for the various delivery mechanisms:

Application Server Product Release	Delivery Mechanism
Application Server Enterprise Edition component within the Sun Java Enterprise System.	File-based distribution patch installation needed through SunSolve

J2EE Support

The Sun Java System Application Server 8.1 2005Q2 supports the J2EE 1.4 platform. The following table describes the enhanced APIs available on the J2EE 1.4 platform.

Table 2 Major API changes on the J2EE 1.4 Platform

API	Description
Components	
Application and Application Client	Implementation of standard deployment descriptors by means of XML schemas
Enterprise JavaBeans (EJB) 2.1	Timer service and EJB Web-service endpoint
Java Servlet 2.4	Web-service endpoint filter
JavaServer Pages (JSP) 2.0 architecture	Expression language and tag library
J2EE Connector Architecture 1.5	Inbound resource adaptor and Java Message Service (JMS) pluggability
Web Services	
Java Web Services Developer Pack 1.5	Integrated toolkit for building, testing and deploying XML applications, Web services, and Web applications
Java API for XML-based Remote Procedure Calls (JAX-RPC) 1.1	Mapping for WSDL and Java technology and support for development of Web-service clients and endpoints
WS-I Basic Profile 1.0	The enabling element for interoperability using WSDL and SOAP
SOAP with attachment API for Java (SAAJ) 1.2	An API for SOAP-based messaging; fosters the creation of SOAP messages with attachments
Java APIs for XML Registries (JAXR) 1.0	A uniform and standard API for accessing XML registries, such as those for Universal Description Discovery and Integration (UDDI and ebXML)
Other	
J2EE Deployment 1.1	Standard APIs that enable deployments of J2EE components and applications
J2EE Management 1.0	Definitions for the information model for managing the J2EE platform

Table 2 Major API changes on the J2EE 1.4 Platform (*Continued*)

API	Description
Java Management Extensions (JMX) 1.2	Standard management API
Java Authorization Contract for Containers (JACC) 1.0	Definitions of security contracts between a J2EE Application Server and the authorization policy provider
Java API for XML Processing (JAXP) 1.2	An API with which applications can parse and transform XML documents; also adds support for processing of XML schemas
JMS 1.1	A messaging standard that enables J2EE application components to create, send, receive, and read messages; also adds support for uniform APIs for queues and topics
JavaMail 1.3	A set of abstract classes that model a mail system; also includes minor updates to the APIs

High Performance

The Application Server includes a high performance EJB container, Web container and services, and supports concurrent message delivery with the Sun Java System Message Queue software.

Scalability

The Application Server supports horizontal scalability through clustering of server instances and request load balancing. It also achieves class leading vertical scalability supporting large multi-processor machines. The integrated message broker can be clustered for better scalability and availability. Client access from HTTP clients, RMI/IIOP based Rich Client Applications, Web Services Clients, and JRM Clients can be load balanced to Application Server clusters.

High Availability

The Application Server includes load balancing for HTTP, IIOP, and JMS clients; HTTP session failover support; EJB clustering and failover support; highly available EJB timers; distributed transaction recovery; support for rolling application upgrades; and a high availability database for storing the transient state of J2EE applications.

Availability allows for failover protection of Application Server instances in a cluster. If one Application Server instance goes down, another Application Server instance takes over the sessions that were assigned to the unavailable server. Session information is stored in the HADB. HADB supports the persistence of HTTP sessions, Stateful Session Beans, and Single Sign On credentials.

JavaServer Faces 1.1 Support

The Sun Java System Application Server Enterprise Edition 8.1 supports JavaServer Faces 1.1 technology. The JavaServer Faces technology consists of a set of server-side APIs that represent user-interface components that manage their state, event, handling, and input validation. The APIs also define page navigation and support internationalization and accessibility. You can add custom UI components with a JSP custom tag library.

While developing with JavaServer Faces technology, each member of a development team can focus on a single piece of the process. A simple programming model then links the pieces, resulting in a much more efficient and simpler development cycle.

Hardware and Software Requirements

This section lists the requirements that must be met before installing the Sun Java System Application Server Enterprise Edition 8.1 product.

- [Platform Requirements](#)
- [JDBC Drivers and Databases](#)
- [Configuring Oracle](#)
- [Configuring PointBase](#)
- [Web Servers](#)
- [Browsers](#)
- [High Availability Requirements and Limitations](#)
- [Other Requirements](#)

Platform Requirements

The following table lists the operating systems that are supported for Sun Java System Application Server Enterprise Edition 8.1 2005Q2 product. Additionally, the minimum and recommended memory requirements are identified for installing and running the Application Server

Table 3 Sun Java System Application Server 8.1 2005Q2 Platform Requirements

Operating System	Minimum Memory	Recommended Memory	Minimum Disk Space	Recommended Disk Space	JVM
Microsoft Windows 2000 SP4	512 MB	1 GB	250 MB free	500 MB free	JDK 1.5.0.04

Table 3 Sun Java System Application Server 8.1 2005Q2 Platform Requirements (*Continued*)

Operating System	Minimum Memory	Recommended Memory	Minimum Disk Space	Recommended Disk Space	JVM
Microsoft Windows 2003 Enterprise Server	1 GB	2 GB	250 MB free	500 MB free	JDK 1.5.0.04
Microsoft Windows XP	1 GB	2 GB	250 MB free	500 MB free	JDK 1.5.0.04

To check your operating system version, use the `ver` command. To check the disk space use the `mem` command.

JDBC Drivers and Databases

The Sun Java System Application Server is designed to support connectivity to any DBMS with a corresponding JDBC driver. For a list of components that Sun has tested and found to be acceptable for constructing J2EE compatible database configurations, please refer to the following table:

Table 4 JDBC Drivers and Databases

JDBC Vendor	JDBC Driver Type	Supported Database Server
i-net Software	Type 4	Oracle (R) 8.1.7, 9i, 9.2.0.3 Sybase ASE 12.5.2 Microsoft SQL Server 2000 4.0 Service Pack 1
IBM	Type 2	IBM DB2 8.1 Service Pack 3+
PointBase	Type 4	PointBase Network Server 4.8
DataDirect	Type 4	Oracle (R) 8.1.7, 9i, 9.2.0.3 Sybase ASE 12.5.2 Microsoft SQL Server IBM DB2 8.1 Service Pack 3+
Sun Java System JDBC Driver for Oracle	Type 4	Oracle (R) 9.2.0.3, 10G
Sun Java System JDBC Driver for DB2	Type 4	IBM DB2 8.1 Service Pack 3+
Sun Java System JDBC Driver for Sybase	Type 4	Sybase ASE 12.5.2
Sun Java System JDBC Driver for Microsoft SQL Server	Type 4	Microsoft SQL Server 2000 4.0 Service Pack 1
Oracle	Type 4, Type 2	Oracle (R) 9.2.0.3, 10G

For more information about i-net Software, see:

<http://www.inetsoftware.de/>

The following table identifies additional supported JDBC drivers; however these drivers are not J2EE compatible.

Table 5 JDBC Drivers not J2EE compatible

JDBC Vendor	JDBC Driver Type	Supported Database Server
Oracle	Type 4	Oracle (R) 9.2.0.3, 10G
Sybase	jConnector	Sybase ASE 12.5.1

Additional drivers have been tested to meet the JDBC requirements of the J2EE 1.4 platform with the JDBC Driver Certification Program. These drivers can be used for JDBC connectivity with the Sun Java System Application Server. While Sun offers no product support for these drivers, we support the use of these drivers with the Sun Java Enterprise System Application Server.

Configuring Oracle

Oracle JDBC drivers must be configured properly to be compliant with J2EE 1.4. Use the following configuration for Type 2 and Type 4 drivers:

1. Use the JDBC driver from 9.2.0.3 or later.
2. The Oracle database needs to have `compatible=9.0.0.0.0` or higher in its parameter (`init.ora`) file.
3. Use the `ojdbc14.jar` file.
4. Configure the Application Server to define the following JVM property:

```
-Doracle.jdbc.J2EE13Compliant=true
```

In addition, for Type-2 drivers, both the `ORACLE_HOME` and `PATH` variables (which must include `$ORACLE_HOME/lib`) need to be defined in the environment in which the Application Server is started. For example, add them to the `asenv.conf` file and ensure they are exported.

Configuring PointBase

Many sample applications use the PointBase database server included with the Application Server. When using Application Server Enterprise Edition, you must configure the PointBase database server before using it. Before using PointBase with the Application Server, however, note the supported configuration combination.

Table 6 Supported J2SE/PointBase Combinations

Application Server	PointBase
Supported	
J2SE 5.0	J2SE 1.4.2
Unsupported	
J2SE 5.0	J2SE 5.0
J2SE 1.4	J2SE1.4

There are two ways to configure PointBase:

- Set the `JAVA_HOME` environment variable to the location of the J2SE. The PointBase implementation bundled with Application Server 8.1 is only supported with J2SE 1.4.2.
- Edit the Application Server's PointBase configuration file.

To use the first method:

1. Make sure you have the J2SE installed that you want to use.
Download J2SE 1.4.2 if you do not already have it.
2. Using the command appropriate for your operating system and shell, set the `JAVA_HOME` environment variable to the directory in which J2SE is installed; for example:

```
set JAVA_HOME="<JDK1.5_INSTALLDIR>"
```

To use the second method, the procedure depends on the operating system.

Solaris and Linux

Edit the `install_dir/pointbase/tools/serveroption/pbenv.conf` configuration file, changing the line:

```
PB_JAVA=%%%PB_JAVA%%%
```

to

```
PB_JAVA=J2SE_location
```

where `J2SE_location` is the directory where the J2SE is installed. If you installed J2SE with Application Server, it is installed by default to `install_dir/jdk`. After making this change, you can start PointBase using the `startserver` script.

Windows

Edit the `install_dir\pointbase\tools\serveroption\pbenv.bat` configuration file, changing the line:

```
set PB_JAVA=%%%PB_JAVA%%%
to
set PB_JAVA=J2SE_location
```

where `J2SE_location` is the directory in which the J2SE is installed. If you installed J2SE with Application Server, it is installed by default to `install_dir\j2se1.4`. After making this change, you can start PointBase by running `startserver.bat`.

Web Servers

This section lists the web servers that are supported for the Sun Java System Application Server Enterprise Edition 8.1 2005Q2.

Table 7 Supported Web Servers

Web Server	Version	Operating System
Sun Java System Web Server	6.1 +	Solaris SPARC 8, 9, 10 Solaris x86 9, 10 Red Hat Enterprise Linux 2.1 Update 2, 3.0 Update 1

Browsers

This section lists the browsers that are supported with the Sun Java System Application Server Enterprise Edition 8.1 2005Q2.

Table 8 Browsers Supported

Browser	Version
Mozilla	1.4, 1.5, 1.6, 1.7.x
Netscape Navigator	4.79, 6.2, 7.0
Internet Explorer	5.5 Service Pack 2, 6.0

High Availability Requirements and Limitations

The following high availability requirements must be met before configuring the Sun Java System Application Server High Availability component:

- HADB requires 512 Mbytes minimum memory and 1 Gbytes recommended memory to work properly with the Application Server.

- HADB supports IPv4 only.
- The network must be configured for UDP multicast.
- The new HADB management system may show problems handling eight or more hosts.

Other Requirements

The following additional requirements should be met before installing the Sun Java System Application Server software.

- **Free space:** your temporary directory must have a minimum of 300 Mbytes free for Sun Java System Application Server installation, and 250 Mbytes of free space for the SDK installation.
- **Using the uninstall program:** If you need to remove the application server from your system, it is important to use the uninstall program that is included with the software. If you attempt to use another method, problems will arise when you try to reinstall the same version, or when you install a new version.
- **Free ports:** You must have seven unused ports available.
 - The installation program automatically detects ports in use and suggests currently unused ports for the default settings. By default, the initial default ports are 8080 (for admin instance) and 38080 (for AppServer1 instance) for the HTTP server, and 4850 for the Admin Server.
 - The installation program will detect used ports and assign two others for you: Sun Java™ System Message Queue (by default, 7679), and IIOP (by default, 3750 for IIOP and 3347 and 3360 for IIOP/SSL). If these default port numbers are in use, the installation program will assign a random port number from the dynamic port range (note that this may not be the next available port number).
- **Starting previously-installed servers:** Unless you are replacing the previously installed server, you should start it before you begin the Sun Java System Application Server 8.1 installation process. This allows the installation program to detect ports that are in use and avoid assigning them for other uses.
- **Shutting down firewall:** You must stop any firewall before installing the Sun Java System Application Server software, because some of this software disables all ports by default. The installation program must be able to accurately determine which ports are available.

Related Documentation

The Sun Java System Application Server documentation set covers the following product releases:

- Application Server Enterprise Edition 8.1 2005Q1
- Application Server Enterprise Edition 8.1 2005Q2
- Application Server Enterprise Edition 8.1 2005Q2 Update 2

Unless otherwise specified, whenever either product release is mentioned in the documentation, both releases are implied.

In addition to these release notes, the Application Server component includes an entire set of documentation that can be found at this location:

<http://docs.sun.com/app/docs/coll/1310.1>

The following table summarizes the books included in the Application Server core application documentation set.

Table 9 Books in This Documentation Set

Book Title	Description
<i>Release Notes</i>	Late-breaking information about the software and the documentation. Includes a comprehensive, table-based summary of the supported hardware, operating system, JDK, and JDBC/RDBMS.
<i>Quick Start Guide</i>	How to get started with the Sun Java System Application Server product.
<i>Installation Guide</i>	Installing the Sun Java System Application Server software and its components.
<i>Deployment Planning Guide</i>	Evaluating your system needs and enterprise to ensure that you deploy Sun Java System Application Server in a manner that best suits your site. General issues and concerns that you must be aware of when deploying an application server are also discussed.
<i>Developer's Guide</i>	Creating and implementing Java™ 2 Platform, Enterprise Edition (J2EE™ platform) applications intended to run on the Sun Java System Application Server that follow the open Java standards model for J2EE components and APIs. Includes general information about developer tools, security, assembly, deployment, debugging, and creating lifecycle modules.
<i>J2EE 1.4 Tutorial</i>	Using J2EE 1.4 platform technologies and APIs to develop J2EE applications and deploying the applications on the Sun Java System Application Server.
<i>Administration Guide</i>	Configuring, managing, and deploying the Sun Java System Application Server subsystems and components from the Administration Console.
<i>High Availability Administration Guide</i>	Post-installation configuration and administration instructions for the high-availability database.
<i>Administration Reference</i>	Editing the Sun Java System Application Server configuration file, <code>domain.xml</code> .
<i>Upgrade and Migration Guide</i>	Migrating your applications to the new Sun Java System Application Server programming model, specifically from Application Server 6.x and 7. This guide also describes differences between adjacent product releases and configuration options that can result in incompatibility with the product specifications.

Table 9 Books in This Documentation Set (*Continued*)

Book Title	Description
<i>Performance Tuning Guide</i>	Tuning the Sun Java System Application Server to improve performance.
<i>Troubleshooting Guide</i>	Solving Sun Java System Application Server problems.
<i>Error Message Reference</i>	Solving Sun Java System Application Server error messages.
<i>Reference Manual</i>	Utility commands available with the Sun Java System Application Server; written in manpage style. Includes the <code>asadmin</code> command line interface.

Bugs Fixed in This Release

The following table describes the bugs fixed in Application Server 8.1 2005Q2.

Table 10 Fixed Bugs in Application Server 8.1 2005Q2

Bug Number	Description
6295958	<p>Impossible to enter AS master password and the master pswd is unknown to the user.</p> <p>Note: Master password will be the same as admin password in the Configure Automatically during Installation mode whereas in the Configure Manually after Installation mode, the user can specify the different password in the CLI.</p>

Important Information

This section covers the following topics:

- [Installation Notes](#)
- [Compatibility Issues](#)
- [Deploytool](#)
- [Verifier](#)
- [Classloader Changes](#)
- [Web Service Security Configuration](#)
- [Accessibility Features for People With Disabilities](#)

Installation Notes

For information about patch requirements and installation, see the following section:

Patch Requirement Information

The following table gives the numbers and minimum versions for the alignment patches. All patches referred to in this section are the minimum version number required for upgrade. It is possible that a new version of the patch has been issued since this document was published. A newer version is indicated by a different version number at the end of the patch. For example: 123456-04 is a newer version of 123456-02 but they are the same patch ID. Refer to the README file for each patch listed for special instructions.

To access the patches, go to <http://sunsolve.sun.com>.

Table 11 Application Server 8.1 2005Q2 Alignment Patches Required For Windows

Patch Number	Patch Description
121533-01	Windows (MSI): Sun Java™ System Message Queue 3 2005Q4
121523-01	Windows (MSI): Shared Components Patch
121528-01	Windows (MSI): Sun Java™ System Application Server 8.1 2005Q2

For detailed information about Upgrade procedure of the Application Server from JES3 to JES4 refer *Sun Java Enterprise System 2005Q4 Upgrade Guide for Microsoft Windows* located at <http://docs.sun.com/app/docs/doc/819-4461>.

Compatibility Issues

In the next major release of the Sun Java System Application Server Enterprise Edition the following incompatibilities will be introduced:

- While the HTTP Service will continue using a DNS cache for better performance, monitoring of the DNS cache will not be available.
- The support for HTTP file caching will be revamped, resulting in changes to configuration and monitoring.

- The format for the access log rotation suffix will be changed to the format supported by date and time objects as specified in <http://java.sun.com/j2se/1.5.0/docs/api/java/text/SimpleDateFormat.html>. The default value in this release, “%YYYY;%MM;%DD;-%hh;h:mm;m:ss;s,” will continue to be supported but no other variations will be supported.
- Any `domain.xml` elements, attributes and properties no longer supported will be flagged as warnings in the server log and in the upgrade log file as having been deprecated.
- The `server.http-service.dns` node will no longer be available in the monitoring view.
- Some of the attributes from the `server.http-service.file-cache` node may be removed. Consequently, any `asadmin` monitoring command trying to access removed attributes from these nodes will fail.

Deploytool

Deploytool will no longer be available. The equivalent functionality is available in the NetBeans IDE. For more information and to plan a migration, please see J2EE 1.4 tutorial for NetBeans 4.1 at <http://www.netbeans.org/kb/41/j2ee-tut/index.html>.

Verifier

- Verifier GUI mode (invoked by `verifier -u`) will no longer be available. The equivalent functionality will be available in the NetBeans IDE.
- The default mode for application verification when using verifier tool will change from “Verify J2EE rules” to “Verify J2EE rules and Sun Application Server Configuration Rules.” In other words, by default verifier will test whether an application meets J2EE rules and is configured to run on Sun Application Server. The verifier command will have a command-line switch to test an application for J2EE rules only.

Classloader Changes

In the current release, the JAR and directory entries added to `classpath-prefix`, `server-classpath`, and `classpath-suffix` attributes of `domain.xml` (application server configuration file) are available in the JVM system classpath. An application depending on this behavior might be using the following methods from the class `java.lang.ClassLoader` to access classes or other resources from JVM system classpath:

- `getSystemClassLoader()`

- `getSystemResource()`
- `getSystemResourceAsStream()`
- `getSystemResources`

In the next major release, the JAR and directory entries added to `classpath-prefix`, `server-classpath`, and `classpath-suffix` will no longer be available in the JVM system classpath. If an application uses one of the methods mentioned above, Sun strongly recommends using an equivalent method that does not assume that the resources are available in the system classpath. The equivalent methods that do not rely on the JVM system classpath are available in `java.lang.ClassLoader` and should be used when possible; for example:

EXAMPLE: Old Code

```
java.net.URL url = ClassLoader.getSystemResource
("com/acme/tools/tools.properties");
```

EXAMPLE: Suggested Change

```
java.net.URL url = this.getClass().getClassLoader().getResource
("com/acme/tools/tools.properties");
```

If it is not possible to change the code, then you may choose to use a new configuration option that will be added in the next release to set JVM system classpath.

Web Service Security Configuration

Security for Web services can be configured using the files `wss-client-config.xml` and `wss-server-config.xml`. Please note that the content and names of these configuration files are unstable and likely to change. The equivalent functionality will continue to be available.

Accessibility Features for People With Disabilities

To obtain accessibility features that have been released since the publishing of this media, consult Section 508 product assessments available from Sun upon request to determine which versions are best suited for deploying accessible solutions. Updated versions of applications can be found at: <http://sun.com/software/javaenterprisesystem/get.html>.

For information on Sun's commitment to accessibility, visit <http://sun.com/access>.

Known Issues and Limitations

This section describes the known issues and limitations of Application Server Enterprise Edition 8.1 2005Q2 for Microsoft Windows.

This section describes known problems and associated workarounds for the Sun Java System Application Server Enterprise Edition 8.1 2005Q2 component. If a summary statement does not specify a particular platform, the problem applies to all platforms. This information is organized into the following sections:

- [Administration](#)
- [Application Client](#)
- [Bundled Sun JDBC Drivers](#)
- [Connectors](#)
- [Documentation](#)
- [Logging](#)
- [Monitoring](#)
- [PointBase](#)
- [Samples](#)
- [Security](#)
- [Load Balancer](#)
- [Configuration](#)

Administration

This section describes known administration issues and associated solutions.

Bug ID

6196993

Summary

Cannot restore backed-up domain with another name.

Mirroring of a domain on the same Application Server installation cannot be performed using the `backup-domain` and `restore-domain` commands because the domain cannot be restored using a different name than the original, even though the `asadmin restore-domain` command provides an option to rename the domain. Renaming the backed-up domain appears to succeed, but attempts to start the renamed domain fail because the entries in the domain configuration are not changed, and `startserv` and `stopserv` use the original domain name to set paths.

Solution

The domain name used for `restore-domain` must be the same as that used for the original `backup-domain` command. The `backup-domain` and `restore-domain` commands in Application Server 8.1 work only for backing up and restoring the same domain on the same machine.

Bug ID

6200011

Summary

Starting Application Server with additional JMX Agent is not supported.

J2SE 1.4.x, 5.0, or later can be configured on the Application Server. An integral feature of J2SE 5.0 platform is the ability to start a JMX agent. This is activated when you explicitly set system properties at the server startup.

Example values include:

```
name="com.sun.management.jmxremote" value="true"
```

```
name="com.sun.management.jmxremote.port" value="9999"
```

```
name="com.sun.management.jmxremote.authenticate" value="false"
```

```
name="com.sun.management.jmxremote.ssl" value="false"
```

After configuring JMX properties and starting the server, a new `jmx-connector` server is started within the Application Server Virtual Machine. An undesirable side-effect of this is that the administration functions are affected adversely, and the Application Server administration Console and command—line interface may produce unexpected results. The problem is that there are some conflicts between the built in `jmx-connector` server and the new `jmx-connector` server.

Solution

If using `jconsole` (or any other JMX-compliant client), consider reusing the standard JMX Connector Server that is started with Application Server startup.

When the server starts up, a line similar to the one shown below appears in the `server.log`. You can connect to the JMXService URL specified there and perform the same management/configuration operations after successfully providing the credentials; for example:

```
[#|2004-11-24T17:49:08.203-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system
```

6236544, 6275436

Load balancer configuration file does not get created with the endpoint URL of any web service.

When setting up the load balancer configuration with an application that has an EJB module that exports a web service URL, the context root for the web service isn't in the resulting `loadbalancer.xml` file.

Solution

1. Edit the `loadbalancer.xml` file to add the missing web module as follows:

```
<web-module context-root="context-root-name"
disable-timeout-in-minutes="30" enabled="true"/>
```

2. Replace `context-root-name` value with the context root name of the web service that was exposed as an EJB.

Application Client

This section describes known application client issues and associated solutions.

Bug ID	Summary
6193556	<p>Library JAR packaged in Application Client Archive overwrites MANIFEST file.</p> <p>If you have a top level JAR file inside your client JAR (in this case, reporter.jar), when you deploy the client JAR, the MANIFEST file for that JAR overwrites the MANIFEST file for the client JAR.</p> <p><i>Solution</i></p> <p>None at this time.</p>

Bundled Sun JDBC Drivers

This section describes known bundled Sun JDBC driver issues and associated solutions.

Bug ID	Summary
6165970	<p>Applications using the <code>TRANSACTION_SERIALIZABLE</code> isolation level with the bundled Sun driver for Microsoft SQL Server may hang when using a prepared statement to update if two parallel transactions are running and one of them is rolled back.</p> <p>To set a desired isolation level for a connection, the corresponding connection pool must be created at that same isolation level. See the Administration Guide for details about configuring connection pools.</p> <p><i>Solution</i></p> <p>None at this time.</p>

Bug ID

6170432

Summary

PreparedStatement Errors.

Description #1

If an application generates more than 3000 PreparedStatement objects in one transaction, the following error may occur with DB2:

```
[sunm] [DB2 JDBC Driver] No more available statements.Please recreate your package with a larger dynamicSections value.
```

Solution #1

Add following properties to the connection pool definition to get the driver to rebind DB2 packages with a larger dynamic sections value:

```
createDefaultPackage=true replacePackage=true  
dynamicSections=1000
```

See the *Administration Guide* for details about configuring connection pools.

Description #2

Related to the PreparedStatement error above, another error message that may be thrown is:

```
[sunm] [DB2 JDBC Driver] [DB2]Virtual storage or database resource is not available.
```

Solution #2

Increase the DB2 server configuration parameter *APPLHEAPSZ*. A good value is 4096.

Description #3

Isolation level TRANSACTION_SERIALIZABLE. If your application uses isolation level TRANSACTION_SERIALIZABLE and uses one of the parameters suggested above, it might hang while obtaining a connection.

Solution #3

To set desired isolation level for a connection, the corresponding connection pool has to be created at that isolation level. See the *Administration Guide* for instructions.

Bug ID

6189199

Summary

Problems setting isolation level with the bundled Sun driver for Sybase Adaptive Server.

Applications using the TRANSACTION_SERIALIZABLE isolation level with the bundled Sun driver for Sybase Adaptive Server may hang when using a prepared statement to update if two parallel transactions are running and one of them is rolled back. Connection rollback fails with following message, and the rolled back connections cannot be used anymore:

```
java.sql.SQLException: [sunm] [Sybase JDBC Driver]Request cannot be
submitted due to wire contention
```

Sybase Adaptive Server does not support the TRANSACTION_REPEATABLE_READ isolation level. However, querying DatabaseMetaData, the bundled Sun driver returns that this isolation level is supported by the database. Applications using the this isolation level will fail.

Applications using the bundled Sun driver cannot set the TRANSACTION_READ_UNCOMMITTED isolation level. The application throws the following exception on the first DataBaseMetaData access:

```
java.sql.SQLException: [sunm] [Sybase JDBC Driver] [Sybase]The optimizer
could not find a unique index which it could use to perform an
isolation level 0 scan on table 'sybssystemprocs.dbo.spt_server_info'.
```

Solution

None at this time.

Connectors

This section describes known J2EE connector architecture issues and associated solutions.

Bug ID

6188343

Summary

After restarting a DAS instance, undeploying the connector module fails when cascade is set to false.

In this scenario, a standalone or embedded connector module is deployed in DAS and connector connection pools, and resources are created for the deployed module. After restarting the DAS instance, undeploying the connector module fails when cascade is set to false with the following exception:

```
[#|2004-10-31T19:52:23.049-0800|INFO|sun-appserver-ee8.1|javax.enterp
rise.system .core|_ThreadID=14;|CORE5023: Error while unloading
application [foo]|#]
```

Solution

Use cascaded undeploy (set the cascade option to true) for undeploying standalone and embedded connectors after restart of the DAS instance.

Documentation

This section describes known documentation issues and associated solutions.

Bug ID	Summary
Various IDs	<p>Javadoc Inconsistencies.</p> <p>The Javadoc for several AMX interfaces and methods is either missing or incorrect:</p> <p>Getter methods for <code>NumConnAcquired</code> and <code>NumConnReleased</code> statistics are missing from <code>ConnectorConnectionPoolStats</code> and <code>AltJDBCConnectionPoolStats</code>. These getter methods will be added in a future release as <code>getNumConnAcquired()</code> and <code>getNumConnReleased()</code>.</p> <p>Calling the following methods in <code>EJBCacheStats</code> will throw an exception: <code>getPassivationSuccesses()</code>, <code>getExpiredSessionsRemoved()</code>, <code>getPassivationErrors()</code>, <code>getPassivations()</code>. This will be fixed in a future release.</p> <p>The AMX MBeans may require several seconds after server startup before they are all registered and available for use. A future release will make it possible to determine when the AMX MBeans are fully loaded.</p> <p>The constant <code>XTypes.CONNECTOR_CONNECTION_POOL_MONITOR</code> is misspelled ("NNN"). This will be corrected in a future release.</p>
6265624	<p>Bundled ANT throws <code>java.lang.NoClassDefFoundError</code>.</p> <p>The following exception is thrown in thread "main"</p> <pre>java.lang.NoClassDefFoundError: org/apache/tools/ant/launch/Launcher.</pre> <p>Solution</p> <p>Use the bundled ANT for things outside the Application Server is not recommended.</p>

Lifecycle Management

This section describes known lifecycle management issues and associated solutions.

Bug ID

6193449

Summary

After setting the `ejb-timer-service` property `minimum-delivery-interval` to 9000, an attempt to set the `ejb-timer-service` property `redelivery-interval-in-millis` to 7000 causes the set command to fail with the following error:

```
[echo] Doing admin task set
[exec]
[Attribute(id=redelivery-interval-internal-in-millis) :
Redelivery-Interval should be greater than or equal to
Minimum-delivery-interval-in-millis (9,000)]
[exec] CLI137 Command set failed.
```

`minimum-delivery-interval` is the minimal interval duration between deliveries of the same periodic timer.

`redelivery-interval-in-millis` is the time the timer service will wait after a failed `ejbTimeout` before attempting redelivery.

The problem is that the logic that relates the redelivery interval property to the minimum delivery property is incorrect and prevents you from using the GUI or the CLI to set any value where the minimum delivery interval is greater than redelivery interval.

The `minimum-delivery-interval-in-millis` must always be set equal to or higher than `ejb-timer-service` property `redelivery-interval-in-millis`. The problem is that there is an erroneous validation check in the Application Server to verify that the value for `redelivery-interval-in-millis` is greater than the value for `minimum-delivery-interval-in-millis`.

Solution

Use the default values for these properties, as follows:

```
minimum-delivery-interval (default)=7000
redelivery-interval-in-millis (default)=5000
```

Values other than these defaults will generate an error.

Logging

This section describes known logging issues and solutions.

Bug ID	Summary
6180095	<p>Setting debug statement for <code>access, failure</code> causes hanging in Application Server startup.</p> <p>Setting the <code>java.security.debug</code> option for the JVM will cause the server instance startup to freeze with a deadlock; for example, setting the following in <code>domain.xml</code> causes the problem:</p> <pre><jvm-options>-Djava.security.debug=access, failure</jvm-options></pre> <p>None at this time. Please avoid setting this flag.</p>

Monitoring

This section describes known monitoring issues and associated solutions.

Bug ID	Summary
6174518	<p>Some of the HTTP Service monitoring statistics do not present useful information and should be ignored.</p> <p>When viewing the monitoring statistics of some elements of the HTTP Service, some values presented do not correspond to current values or are always 0. Specifically, the following HTTP Service statistics do not present information applicable to the Application Server, and should be ignored:</p> <p><code>http-service</code></p> <ul style="list-style-type: none"><code>load1MinuteAverage</code><code>load5MinuteAverage</code><code>load15MinuteAverage</code><code>rateBytesTransmitted</code><code>rateBytesReceived</code> <p><code>pwc-thread-pool</code> (the element)</p> <p><i>Solution</i></p> <p>These monitors will be removed in future releases and replaced with more appropriate information.</p>

Bug ID

6191092

Summary

Monitoring MBean for an undeployed EJB module is not removed, even though all statistics under that monitoring name are moved.

For example:

```
EJBModuleMonitorMap().size() = 1 eventhough ejb module is undeployed
EJBModu
```

This true for both EJB modules and applications. Both programmatically (through MBean API) and through asadmin list/get, an empty monitoring MBean still exists.

Diagnostics

asadmin list -m "server.applications" shows the following output:

```
server.applications.MEjbApp
server.applications.__ejb_container_timer_app
server.applications.adminapp
server.applications.admingui
server.applications.com_sun_web_ui
server.applications._export_install_nov-11_domains_domain1_applications_j2ee
-
```

You can look at statistics:

```
bin/asadmin list -m "server.applications._expo
rt_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01
"
server.applications._export_install_nov-11_domains_domain1_applications_j2ee
ules_
sqe_ejb_s1_01.SQEMessage
server.applications._export_install_nov-11_domains_domain1_applications_j2ee
ules_
sqe_ejb_s1_01.TheGreeter
```

Once you undeploy:

```
_export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_
s1_0
```

If you do a list command, you still see the application:

Bug ID

Summary

```
asadmin list -m "server.applications"  
server.applications.MEjbApp  
server.applications.__ejb_container_timer_app  
server.applications._export_install_nov-11_domains_domain1_applications_j2ee  
ules_sqe_ejb_s1_01  
server.applications.adminapp  
server.applications.admingui  
server.applications.com_sun_web_ui
```

but it does not contain any monitoring statistics:

```
asadmin list -m "server.applications._expo  
rt_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01  
"
```

Nothing to list at server.applications.-export-install-nov-11-domains-domain1 applications-j2ee-modules-sqe-ebb-s1-01.

To get the valid names beginning with a string, use the wildcard ("**") character. For example, to list the names of all the monitorable entities that begin with server, use list "server.*".

Solution

This is harmless. Module can be safely redeployed with out any problems. The root monitoring Mbean is not removed, but it is empty.

PointBase

This section describes known and associated solutions related to PointBase.

Bug ID

Summary

6184797

Setting the isolation levels on a connection pool for an application causes exceptions in PointBase.

For a JDBC connection pool pointing to a PointBase database installation, setting the transaction-isolation-level pool attribute to any value other than the default (`Connection.TRANSACTION_READ_COMMITTED`) causes an exception. However, setting this same parameter to non-default values for pools pointing to other databases does not throw an exception.

Solution

For a JDBC connection pool pointing to a PointBase database installation, do not attempt to set the transaction-isolation-level.

Bug ID**Summary**

6204925

PointBase throws an exception if a network server and embedded drivers are used together.

The bundled PointBase sometimes throws an exception if the network server driver and the embedded driver are simultaneously used.

Solution

Use either the embedded driver or the network server driver, but not both.

6264969,6275448

Upgrade problem where the default PointBase database is overwritten

When upgrading to Application Server Enterprise Edition 8.1 2005Q2 Update 2, the Update release patch overwrites the Pointbase default database.

Solution

Recreate or re-enter any scheme or data that existed prior to the upgrade. If you deployed applications with CMP beans with the generate table option, you must undeploy or redeploy the application to have the tables regenerated.

Samples

This section describes known and associated solutions related to the sample code included with the Application Server 8.1 product.

Bug ID

6195092

Summary

On Windows, setup-one-machine-cluster hangs but works on Solaris; mqfailover requires Ctrl+C to cancel and then must be re-run.

From

install_dir\samples\ee-samples\failover\apps\mqfailover\docs\index.html, if you run the following commands:

- Console 1

```
cd install_dir\samples\ee-samples asant
start-mq-master-broker1
```

- Console 2

```
cd install_dir\samples\ee-samples asant
start-mq-cluster-broker1
```

- Console 3

```
cd install_dir\samples\ee-samples asant
start-mq-cluster-broker2
```

- Console 4

```
cd install_dir\samples\ee-samples asadmin
start-domain domain1
```

If you have already executed

asant setup-one-machine-cluster-without-ha or asant setup-one-machine-cluster-with-ha for any other Enterprise Edition sample, then execute asant configure-mq otherwise execute

asant setup-one-machine-cluster-and-configure-mq. In this case, the command appears to succeed:

```
start_nodeagent: [echo] Start the node agent cluster1-nodeagent [exec]
Comman
```

But then the system hangs indefinitely.

Solution

None at this time. This problem similarly affects all Enterprise Edition samples that use this ant target on Windows. A workaround is to Ctrl+C out of the hung process and then rerun it.

Bug ID

6198003

Summary

Documentation does not explicitly state that you need to create JMS resources before running the MQ Failover Sample Application following the asadmin deploy instructions.

The error thrown is as follows:

```
Install_Location>\AppServer\domains\domain1\config\sun-acc.xml -name
MQFailoverTestClient -t Nov 18, 2004 10:50:17 PM
com.sun.enterprise.naming.NamingManagerImpl bindObjects SEVERE:
NAM0006: JMS Destination object not found: jms/durable/TopicA Nov 18,
2004 10:50:18 PM com.sun.enterprise.naming.NamingManagerImpl
bindObjects SEVERE: javax.naming.NameNotFoundException
javax.naming.NameNotFoundException
```

The documentation does not explicitly state that JMS resources must be manually created if manual deployment is done using asadmin deploy commands, and that the provided ant targets to deploy the sample application should be used.

Solution

Use the asant deploy target for the build.xml script, which creates the required JMS resources to run the application.

Security

This section describes known issues and associated solutions related to Application Server and web application security and certificates.

Bug ID

6183318

Summary

Cannot run WebServiceSecurity applications on Enterprise Edition with J2SE 5.0.

WebServiceSecurity applications cannot run with J2SE 5.0 because:

- J2SE 5.0 PKCS11 does not support UNWRAP mode
- J2SE 5.0 PKCS11 does not support

RSA/ECB/OAEPWithSHA1AndMGF1Padding with
PKCS11

The J2SE team has filed "CR 6190389: Add support for the RSA-PKCS1 and RSA-OAEP wrap/unwrap mechanisms" for this bug.

Solution

Use J2SE 1.4.2 with any other JCE provider (not the one included by default). Note that hardware accelerator support will not be present in this configuration.

Web Container

This section describes known web container issues and associated solutions.

Bug ID	Summary
5004315	<p data-bbox="318 390 1190 440">On Windows, deploying an application using <code>--precompilejsp=true</code> can lock JAR files in the application, causing later undeployment or redeployment to fail.</p> <p data-bbox="318 458 1222 591">If you request precompilation of JSPs when you deploy an application on Windows, later attempts to undeploy that application or to redeploy it (or any application with the same module ID) will not work as expected. The problem is that JSP precompilation opens JAR files in your application but does not close them, and Windows prevents the undeployment from deleting those files or the redeployment from overwriting them.</p> <p data-bbox="318 609 1222 713">Note that undeployment succeeds to a point, in that the application is logically removed from the Application Server. Also note that no error message is returned by the <code>asadmin</code> utility, but the application's directory and the locked jar files remain on the server. The server's log file will contain messages describing the failure to delete the files and the application's directory.</p> <p data-bbox="318 730 1219 864">Attempts to redeploy the application after undeploying fail because the server tries to remove the existing files and directory, and these attempts also fail. This can happen if you try to deploy any application that uses the same module ID as the originally deployed application, because the server uses the module ID in choosing a directory name to hold the application's files.</p> <p data-bbox="318 881 1210 904">Attempts to redeploy the application without undeploying it first will fail for the same reasons.</p> <p data-bbox="318 921 434 944"><i>Diagnostics</i></p> <p data-bbox="318 961 1203 1012">If you attempt to redeploy the application or deploy it after undeploying it, the <code>asadmin</code> utility returns an error similar to the one below.</p> <p data-bbox="318 1029 1186 1052">An exception occurred while running the command. The exception message is: CL</p> <p data-bbox="318 1069 401 1091"><i>Solution</i></p> <p data-bbox="318 1109 1226 1192">If you specify <code>--precompilejsps=false</code> (the default setting) when you deploy an application, then this problem will not occur. Be aware that the first use of the application will trigger the JSP compilation, so the response time to the first request will be longer than for later requests.</p> <p data-bbox="318 1209 1222 1288">Note also that if you do precompile, you should stop and restart the server before undeploying or redeploying the application. The shutdown frees the locked JAR files so the undeployment or redeployment after the restart can succeed.</p>

Bug ID

6172006

Summary

Unable to deploy WAR with Servlet 2.4-based web.xml that contains an empty `<load-on-startup>` element.

The optional load-on-startup servlet element in a web.xml indicates that the associated servlet is to be loaded and initialized as part of the startup of the web application that declares it.

The optional content of this element is an integer indicating the order in which the servlet is to be loaded and initialized with respect to the web application's other servlets. An empty `<load-on-startup>` indicates that the order is irrelevant, as long as the servlet is loaded and initialized during the startup of its containing web application.

The Servlet 2.4 schema for web.xml no longer supports an empty `<load-on-startup>`, meaning that an integer must be specified when using a Servlet 2.4 based web.xml. If specifying an empty `<load-on-startup>`, as in `<load-on-startup/>`, the web.xml will fail validation against the Servlet 2.4 schema for web.xml, causing deployment of the web application to fail.

Backwards compatibility issue. Specifying an empty `<load-on-startup>` still works with Servlet 2.3 based web.xml.

Solution

Specify `<load-on-startup>0</load-on-startup>` when using a Servlet 2.4 based web.xml to indicate that servlet load order does not matter.

Bug ID

6184122

Summary

Unable to compile JSP page on resource constrained servers.

The JSP page is accessed but fails to compile, and the server log contains the error message "Unable to execute command" with the following stack trace:

```
at org.apache.tools.ant.taskdefs.Execute$Java13CommandLauncher.exec(Execute.j
at org.apache.tools.ant.taskdefs.Execute.execute(Execute.java:427)
at org.apache.tools.ant.taskdefs.compilers.DefaultCompilerAdapter.executeExte
at org.apache.tools.ant.taskdefs.compilers.JavacExternal.execute(JavacExterna
at org.apache.tools.ant.taskdefs.Javac.compile(Javac.java:842)
at org.apache.tools.ant.taskdefs.Javac.execute(Javac.java:682)
at org.apache.jasper.compiler.Compiler.generateClass(Compiler.java:396)
```

Solution

Set the JSP compilation switch "fork" to "false."

This can be done either of two ways:

- Globally, by setting the fork init parameter of the JspServlet in `$(S1AS_HOME)/domains/domain1/config/default-web.xml` to false:

```
<servlet> <servlet-name>jsp</servlet-name>
<servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class> .... <
<param-name>fork</param-name> <param-value>>false</param-value> </init-para
```

- On a per-web application basis, by setting the fork JSP configuration property in `sun-web.xml` to false:

```
<sun-web-app> <jsp-config> <property name="fork" value="false" /> </jsp-co
```

Either setting will prevent ant from spawning a new process for javac compilation.

Bug ID

6188932

Summary

Application Server does not support auth-passthrough Web Server 6.1 Add-On.

The Sun Java System Application Server Enterprise Edition 8.1 2005Q2 Update 2 adds support for the functionality provided by the `auth-passthrough` plugin function available with Sun Java System Application Server Enterprise Edition 7.1. However, in Application Server Enterprise Edition 8.1 2005Q2 Update 2, the `auth-passthrough` plugin feature is configured differently.

The `auth-passthrough` plugin function in Application Server Enterprise Edition 7.1 has been useful in two-tier deployment scenarios, where:

- Application Server instance is protected by a second firewall behind the corporate firewall.
- No client connections are permitted directly to the Application Server instance.

In such network architectures, a client connects to a front-end web server, which has been configured with the `service-passthrough` plugin function and forwards HTTP requests to the proxied Application Server instance for processing. The Application Server instance can only receive requests from the web server proxy, but never directly from any client hosts. As a result of this, any applications deployed on the proxied Application Server instance that query for client information, such as the client's IP address, will receive the proxy host IP, since that is the actual originating host of the relayed request.

In Application Server Enterprise Edition 7.1, the `auth-passthrough` plugin function could be configured on the proxied Application Server instance in order to make the remote client's information directly available to any applications deployed on it; as if the proxied Application Server instance had received the request directly, instead of via an intermediate web server running the `service-passthrough` plugin.

In Application Server Enterprise Edition 8.1 2005Q2 Update 2, the `auth-passthrough` feature may be enabled by setting the `authPassthroughEnabled` property of the `<http-service>` element in `domain.xml` to `TRUE`, as follows:

```
<property name="authPassthroughEnabled" value="true"/>
```

The same security considerations of the `auth-passthrough` plugin function in Application Server Enterprise Edition 7.1 also apply to the `authPassthroughEnabled` property in Application Server Enterprise Edition 8.1 2005Q2 Update 2. Since `authPassthroughEnabled` makes it possible to override information that may be used for authentication purposes (such as the IP address from which the request originated, or the SSL client certificate), it is essential that only trusted clients or servers be allowed to connect to an Application Server Enterprise Edition 8.1 2005Q2 Update 2 instance with `authPassthroughEnabled` set to `TRUE`. As a precautionary measure, it is recommended that only servers behind the corporate firewall should be configured with `authPassthroughEnabled` set to `TRUE`. A server that is accessible through the Internet must never be configured with `authPassthroughEnabled` set to `TRUE`.

Notice that in the scenario where a proxy web server has been configured with the `service-passthrough` plugin and forwards requests to an Application Server 8.1 Update 2 instance with `authPassthroughEnabled` set to `TRUE`, SSL client authentication may be enabled on the web server proxy, and disabled on the proxied Application Server 8.1 Update 2 instance. In this case, the proxied Application Server 8.1 Update 2 instance will still treat the request as though it was authenticated via SSL, and provide the client's SSL certificate to any deployed applications requesting it.

Bug ID	Summary
6314126	If running or installing any Sun Application Server on the Windows platform, it is a requirement that the file system be NTFS and not FAT or FAT32.

Load Balancer

User has to specify the following entries to the CLI while configuring Load Balancer in Configure Manually After Installation mode

The Load-Balancer Plug-in is configured to use ports 1111 and 1112 in the server used for AS_WSINSTANCENAME in the ASConfigurator.properties file. Following entries must be specified to the command line interface before configuring Load Balancer in Configure Manually After Installation mode:

- AS_WSINSTALLDIR= [INSTALLDIR] \\WebServer
- AS_WSINSTANCEDIR= [INSTALLDIR] \\WebServer\\ [INSTANCENAME]
- AS_WSINSTANCENAME= [INSTANCENAME]

where, INSTANCENAME is the current Web server instance name

- AS_LB_PLUGIN_TYPE=Sun ONE Web Server

Workaround

None.

The Load-Balancer Plug-in is configured to use ports 1111 and 1112

The Load-Balancer Plug-in is configured by default to use ports 1111 and 1112 in the ASConfigurator.properties file.

Workaround

None.

Application Server installation behavior and Application Server components do not correspond to JES 3 FCS Solaris/Linux build (6290539)

In Windows, if Application Server is selected, all the subcomponents will get selected by default. This is the behavior of MSI. Users have to deselect the specific sub components that are not needed, like load balancer and node agent.

In Windows Application Server has hard dependency in HADB. So the user can not install Application Server by unselecting the HADB in Configure Automatically During Installation mode. However the user can unselect HADB in Configure Later After Installation mode.

Configuration

Initial configurator in Configure Manually after Installation does not have GUI

The initial configurator used in Configure Manually after Installation mode does not have a GUI support.

Workaround

CLI is provided to the user for configuring in Configure Manually after Installation mode.

User has to accept default setting for nodeagents while configuring Application Server using Configure Manually after Installation (6372244)

Workaround

Accept the default value (nodeagents) for Nodeagent name during configuration in Configure Manually after Installation mode.

Cannot start server with `-server` option (6369978)

Workaround

Change the forward slashes to back slashes of the value of variable `AS_NATIVE_LAUNCHER_LIB_PREFIX` in the `<Application_Server_Home>\config\asenv.bat` file

i.e. change the value `/jre/bin/client` to `\jre\bin\client`

User can use `\jre\bin\server` also according to performance requirements.

In Configure Automatically During Installation, when Application Server is used as container, instance name used for deploying the applications is 'Appserver1'. This is different from Solaris where the instance name used is 'server' (6287671)

Redistributable Files

Sun Java System Application Server Enterprise Edition 8.1 does not contain any files that can be redistributed.

How to Report Problems and Provide Feedback

Use the following resources to handle problems you may encounter with the Application Server product:

- J2EE-INTEREST list: A mailing list for J2EE questions.
<http://archives.java.sun.com/archives/j2ee-interest.html>
- Bug database on Java Developer Connection: To view bugs or to submit a bug, use the Java Developer Connection Bug Parade.
<http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml>
- Java Technology Forums: An interactive message board for sharing knowledge and questions about Java technologies and programming techniques. Use the J2EE SDK forum here for discussions related to the Sun Java System Application Server 8 Platform Edition product.
<http://forum.java.sun.com/>

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. Use the web-based form to provide feedback to Sun:

<http://www.sun.com/hwdocs/feedback>

Please provide the full document title and part number in the appropriate fields. The part number is a seven-digit or nine-digit number that can be found on the title page of the book or at the top of the document. For example, the part number of these Release Notes document is 819-4264-10.

Additional Sun Resources

Useful Sun Java System information can be found at the following Internet locations:

- Application Server product information:
<http://docs.sun.com/app/docs/doc/819-2551>

- Sun Java developer resources:
<http://developer.java.sun.com/>
- Sun Java 2 Platform, Enterprise Edition (J2EE) site:
<http://java.sun.com/j2ee/>
- Application Server product documentation:
<http://docs.sun.com/app/docs/coll/1310.1>
- Sun Microsystems product documentation:
<http://docs.sun.com/>

Copyright © 2006 Sun Microsystems, Inc. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

SUN PROPRIETARY/CONFIDENTIAL.

U.S. Government Rights - Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

Use is subject to license terms.

This distribution may include materials developed by third parties.

Portions may be derived from Berkeley BSD systems, licensed from U. of CA.

Sun, Sun Microsystems, the Sun logo, Java and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries.

Copyright © 2006 Sun Microsystems, Inc. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuels relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plus des brevets américains listés à l'adresse <http://www.sun.com/patents> et un ou les brevets supplémentaires ou les applications de brevet en attente aux Etats - Unis et dans les autres pays.

Propriété de SUN/CONFIDENTIEL.

L'utilisation est soumise aux termes du contrat de licence.

Cette distribution peut comprendre des composants développés par des tierces parties.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie.

Sun, Sun Microsystems, le logo Sun, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays.